LastName:
FirstName:
Machine Number:

Lab exam on Oracle

M2 D&K 2016

Course documents authorized, but all forms of communications strictly forbidden.

Instructions

You must return: (1) the file containing your sql queries (just save the file) (2) this exam sheet (give it to me).

In the sql file:

1. You will write your sql queries in a text file (a plain text file, not word document...) named `dk-2016v1226-lastname.sql`. (replace `lastname` with *your* last name).
2. If you realize your answer is incorrect, you must write so (see example below) or face additional penalties if you fail to mention that your answer is wrong.
3. keep your answers as simple as can be (ex: avoid useless `DISTINCT` statements)
4. Answer questions in order, format answers as illustrated below (indent your answers; i.e., do not write each query on a single line):

```
-- Q1:
SELECT  CASE WHEN 2=3 THEN POWER(3,2)  ELSE 30 END
FROM DUAL;

-- Q2 (incorrect result):
SELECT ...
;
```

On this exam sheet:

- Indicate your name and the number of your working station.
- For each question, write *on this exam sheet* 1 line explaining the main idea of your answer. No need to write a full sentence *here*, keywords are enough; ex:

```
COUNT(*) GROUP BY name
```

1 Creating the database

Load the file `base-northwind.sql` (that file creates and populates the Northwind database).

2 Queries you have to write

Write in SQL the following queries. The result should follow the same schema as the sample answers.

⚠ Queries 1, 2, and 3 are not allowed to use `GROUPING SETS` nor multiple `SELECT` statements.

1. Number of orders for each of the following groups: ship_city, (ship_city, employee), employee and in total. Display the result by increasing order of employee then (for ties) ship_city **(5pt)**

```
 empl_id |    ship_city     | nborder
---------+------------------+---------
       1 | Aachen           |       2
       1 | Albuquerque      |       5
...
       1 |                  |     123
...
       1 | Resende          |       1
...
         |                  |     830
(484 lines)
```

2. Same question, but with the following modifications: **(5+2pt)**
   - add on a fourth column a string indicating the level at which the group is summarized (`city`, `employee`, `city-empl`, or `global`)
   - (bonus) do not display any tuple for the pairs (city, employee) having fewer than 5 orders.

```
 empl_id |    ship_city     | nborder | summary_level
---------+-----------------+---------+---------------
       1 | Albuquerque     |       5 | city-empl
...
       1 |                 |     123 | employee
...
(105 lines for the bonus)
```

3. For each customer country, the list of orders displayed by chronological order (oldest appearing first, most recent appearing last. . .) indicating for each order: **(5+2pt)**

- its total price ($\sum$ unit_price × quantity × (1−discount))
- its ranking in terms of total price (the country's most expensive order(s)) have ranking 1, the next orders have ranking 2. . .)
- you can assume order_id follows chronological order, but get a bonus if you use instead the order_date.

```
   country    | order_id |   price    | ranking
-------------+----------+------------+---------
 Argentina   |    10409 |   319.2000 |       9
 Argentina   |    10448 |   443.4000 |       8
 ...

 Argentina   |    10986 |  2220.0000 |       1
 ...
 Austria     |    10258 |  1614.8800 |      29
 ...
 (830 lines)
```

4. Write an SQL query that computes $\sqrt{12} \times u_{20}$, where $(u_n)_n$ is the sequence defined by **(5pt)**

$$u_0 = 1, \qquad u_{n+1} = u_n + \frac{(-1)^{n+1}}{(2n+3) \times 3^{n+1}}$$

You may find some of the following functions useful:

```
--  MOD:
SELECT MOD(10,2) DROM DUAL; -- 10 modulo 2 = 0
-- SQRT:
SELECT SQRT(26) FROM DUAL; -- square root of 26
--Power:
SELECT POWER(2,3) FROM DUAL; -- 2**3. Second argument should be an integer.
```

```
   the_result
------------------
[An approximation of a well-known number... You have to find out which!]
(1 line)
```